MULTIMEDIA UNIVERSITY

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 3, 2016/2017

TCT 2561 – COMPLEXITY THEORY

(All sections / Groups)

29 MAY 2017
9:00 a.m. – 11:00 a.m.
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This Question paper consists of **5 pages** only including the cover page.
2. Attempt **ALL** questions.
3. All questions carry equal marks and the distribution of the marks for each question is given.
4. Please print all your answers **CLEARLY** in the Answer Booklet provided.

Question 1 (15 marks)

State True or False for each of the following statements. If the answer is false, correct the statement.

(a) Complexity theory only investigates two complexities which are the time complexity and the space complexity.

(b) Complexity theory is to classify solvable problems according to their computational difficulty.

(c) Cryptography is an important practical application of complexity theory.

(d) Certain problems cannot be solved by computers.

(e) The problem of determining whether a mathematical statement is true or false can be solved by computers.

(f) Designing a solution for an NP-complete problem is a creative process. As such, it cannot be reduced to a simple recipe or formula.

(g) The hierarchy theorem and problem solving for space complexity is slightly more difficult than the one for time complexity.

(h) When an optimization problem is NP-hard, no polynomial time algorithm exists that finds the best solution unless P = NP.

(i) It is theoretically proven that parallel computers solve all problems much faster than sequential computers because sequential computers can only do a single operation at a time.

(j) A probabilistic algorithm is an algorithm designed to use the outcome of a deterministic process.

Continued .....

Question 2 (1+2+6+6 marks)

(a) What is the Church-Turing thesis?

(b) What is the relationship between Complexity Theory and Algorithm Analysis?

(c) Prove or disprove the following. Show the necessary steps.
i. $2n^2 = o(n^2)$
ii. $n^3 = \omega(1)$

(d) Consider the following algorithm.

```
 1  string RecurFunc(int anArray[n], int first, int last, int value)
 2  {
 3      if (first > last)
 4          return "Not found"
 5      else
 6      {
 7          int mid = (first + last) / 2
 8          if (value == anArray[mid])
 9              return "Found"
10          else if (value < anArray[mid])
11              return RecurFunc(anArray, first, mid-1, value)
12          else
13              return RecurFunc(anArray, mid+1, last, value)
14      }
15  }
```

i. What does the algorithm compute?
ii. Set up the recurrence relation for the algorithm.
iii. Use the master method to solve the recurrence relation.

Continued .....

Question 3 (9+2+4 marks)

(a) Let $SAT = \{\langle \phi \rangle \mid \phi$ is a satisfiable Boolean formula$\}$.
   i. Give a high-level Turing machine description for SAT that uses exponential time.
   ii. Compute the time needed for each individual stage in the high-level Turing machine description.
   iii. Compute the total time needed in the worst case by the high-level Turing machine description.

(b) Explain the term "space complexity".

(c) Draw one hierarchy that depicts the relationship between the P, NP, NL, coNL, and L complexity classes.

Continued .....

Question 4 (4+4+2+1+4 marks)

A subset of the nodes of a graph G is a dominating set if every other node of G is adjacent to some node in the subset.

Let:

$DOMINATING\text{-}SET = \{\langle G, k\rangle \mid G \text{ has a dominating set with } k \text{ nodes}\}$.

(a) Give a nondeterministic polynomial time Turing machine to prove that *DOMINATING-SET* is in NP.

(b) Give a polynomial time reduction from *VERTEX-COVER* to *DOMINATING-SET*.

(c) Illustrate your reduction with an example by giving an instance of *VERTEX-COVER* graph with k nodes and its corresponding *DOMINATING-SET* graph with k nodes.

(d) Show that your reduction can be done in polynomial time.

(e) Demonstrate that *VERTEX-COVER* graph has a vertex cover with k nodes if and only if the *DOMINATING-SET* graph has a dominating set with k nodes.

End of Page